

基于 GNN 的 IP VPN 的错误配置的检测与定位

专业班级： 信研 2308

姓　　名： 周斐斐

学　　号： 2023200853

基于 GNN 的 IP VPN 的错误配置的检测与定位

摘要

网络（特别是虚拟专用网络）的配置验证是一项复杂的任务，在生产环境的每次更新之前都需要进行，以便网络提供商可以确保其客户的网络可用性。本文探讨了一种基于图神经网络（GNN）的方法，用于检测和定位 IP 虚拟专用网络（VPN）中的配置错误。研究着重于两种 GNN 模型，一种专注于客户和提供商边缘路由器之间的路由配置错误，另一种专注于不同提供商边缘路由器之间的 VPN 路由错误。其目标是提供一种工具，简化验证端到端 VPN 配置的过程。

在研究中，使用了平衡的数据集来训练这两个模型，这个数据集包含了从基于 IMSNetworks 部署的 VPN 中提取的标记配置的示例。结果显示，这两个模型在处理不同规模的 VPN（从 3 到 40 个站点）和两种类型的架构（全网状和中心辐射型）时都表现出很高的准确性。

这种方法的优势在于通过图神经网络可以捕捉网络拓扑和配置之间的复杂关系，从而更有效地检测配置错误。通过使用这种技术，网络提供商可以在每次更新之前对网络配置进行验证，以确保其客户的网络可用性。

关键词：网络管理、配置错误检测、深度学习、图神经网络

Detection And Location Of Misconfiguration Of GNN-Based IP VPN

ABSTRACT

Configuration verification of networks, especially virtual private networks, is a complex task that needs to be done before every update of a production environment so that network providers can ensure network availability for their customers. This paper discusses a graph-based neural network (GNN) approach for detecting and locating configuration errors in IP virtual private networks (VPNS). The study focuses on two GNN models, one that focuses on routing misconfigurations between customer and provider edge routers, and the other on VPN routing misconfigurations between different provider edge routers. The goal is to provide a tool that simplifies the process of verifying an end-to-end VPN configuration.

In the study, both models were trained using a balanced dataset containing examples of tag configurations extracted from an IMSNetwork-based VPN deployment. The results show that both models show high accuracy when dealing with VPNS of different sizes (from 3 to 40 sites) and two types of architectures (full mesh and hub radiant).

The advantage of this method is that the graph neural network can capture the complex relationship between network topology and configuration, so that configuration errors can be detected more effectively. By using this technology, network providers can validate network configurations before each update to ensure network availability for their customers.

KEYWORDS: Network Management, Configuration Error Detection, Deep Learning, Graph Neural Networks

目录

前言

随着信息技术的迅猛发展，企业和组织对于网络安全的关注也日益增强。在众多网络安全威胁中，IP VPN（Internet Protocol Virtual PrivateNetwork）的配置错误可能导致严重的安全隐患和性能问题。

BGP/MPLSIP 虚拟专用网络(VPN)[1]是一种广泛使用的技术，用于互连公司的远程站点并为其提供 Internet 访问。在网络服务提供商内，VPN 的数量的增加会导致配置复杂性的增加，这可能会导致错误，从而影响客户站点的可达性。

防止网络配置错误的传统方法通常需要指定一组模板或一系列形式约束来验证配置的有效性（例如，检查类型正确性）。

确保约束完整性并手动更新它们是很一般的问题。也就是说，我们如何确保验证配置的规则集是完整的？配置是否完全覆盖？从语法和结构的角度来看，配置确实可能是有效的，但对于服务提供商的目标来说是不完整的。例如，忘记在边缘路由器上激活客户的路由表将导致 VPN 丢失站点，尽管其他路由器上的配置是正确的。

IP VPN 是一种通过在公共互联网连接上创建私人网络来实现安全通信的技术。VPN 利用加密和隧道协议确保通过互联网传输的数据的机密性和完整性。企业通常使用 IP VPN 为远程员工提供安全通信，连接分支办事处，并确保敏感数据的机密性。个人用户使用 VPN 进行安全浏览，访问受地区限制的内容以及保持隐私。然而，由于网络拓扑结构复杂、配置参数众多，以及不断变化的业务需求，IP VPN 的配置容易受到影响，从而引发各种问题，比如安全漏洞、性能下降、服务中断等、依赖第三方服务、单点故障等等。

为了有效地识别和纠正这些配置错误，传统的方法通常需要人工分析大量的配置文件和网络日志，耗时长且容易出错。而图神经网络能够学习网络拓扑结构和配置参数之间的复杂关系，通过对网络图的学习，能够在大规模网络中自动发现潜在的配置问题。

本文研究了一种已有的系统[2]，通过图神经网络对 IP VPN 进行深度学习，从而实现对配置错误的实时监测和定位。通过该系统，网络管理员可以更及时地发现潜在的问题并采取相应的措施，从而提高网络的安全性和性能稳定性。

第 1 章 绪论

1.1 研究背景及意义

第 3 层虚拟专用网络是一种网络服务，这是最常见的 VPN 类型，通常称为 IP VPN。它在网络层（第 3 层）上工作，利用互联网在不同地理位置的网络之间建立私密通信通道。IPSec VPN 和 SSL VPN 都属于这一类。

这些 VPN 允许不同客户站点之间通过互联网服务提供商（ISPs）的核心网络互相连接，同时确保每个客户的网络服务质量和数据流隔离。在网络中，有一些提供商边缘（PE）路由器，它们位于 ISPs 核心网络的边缘。每个 PE 路由器连接到一个或多个代表不同客户站点的设备，通常被称为客户边缘（CE）路由器。

提供基于多协议标签交换（MPLS）结构的 VPN 服务需要很高的成本，主要是因为这项服务需要大量的设备和人力投入。首先，需要购买很多路由器和其他网络设备，因为这种服务需要一个强大的网络架构来支持。其次，需要雇佣专业的网络工程师来设计和配置这些设备，这需要花费很多时间和人力成本。此外，这种服务还需要持续的维护和管理，以确保网络的稳定性和安全性，这也是一项长期且持续的成本。总的来说，这种服务是一项高成本、高技术含量的服务，通常只适用于大型企业或机构等有高度需求的客户。对于一般的小型或个人用户来说，可能无法承受这样的成本，也不太可能需要这样的服务。

配置服务提供商的骨干网是一项至关重要且复杂的任务，需要对细节进行精细入微的处理。骨干网作为向多个客户提供可靠和高效连接的基础，其配置涉及在所有骨干路由器上部署一系列协议，包括内部网协议（IGP）、MPLS 和 MP-BGP。

然而，骨干网配置过程的复杂性要求精确实施协议，以在路由器之间实现无缝通信。IGP 在服务提供商网络内传播路由信息，确保选择最佳路径。引入 MPLS 增强了流量工程能力，而 MP-BGP 用于在不同自治系统（AS）之间交换路由信息。尽管涉及复杂性，一旦骨干网配置完成，它往往相对静态，最小化了频繁修改的可能性。

与骨干网相比， VPN 路由和转发表（VRF）表和 CE-PE 路由的配置是服务提供商运营中更为动态且容易出错的方面。不断添加、修改和删除客户站点引入了较高的变异性，使这个过程更容易受到人为错误的影响。每个客户站点的独特需

求要求对细节进行仔细处理，而在参数配置上的简单疏忽可能导致不同站点之间端到端连接的中断。

随着服务提供商网络规模的扩大，容纳越来越多的客户和站点，配置错误的可能性逐渐升高。这些错误可能表现为 VRF 实例的配置错误、CE-PE 路由参数的错误或在处理特定客户流量时的不一致性。这些错误的后果不仅限于个别客户站点，还可能影响整体服务质量和客户满意度。

图 1-1 展示了 BGP/MPLS IP 的简单示例。共包含三个用户，客户 A 有一个全网状 VPN 互连其三个站点（A1、A2 和 A3）。客户 B 有一个中心辐射型 VPN，具有特定的路由策略，允许 B2 和 B3 仅与 B1 通信。最后，客户 C 的 VPN 仅互连两个站点（C1 和 C3）

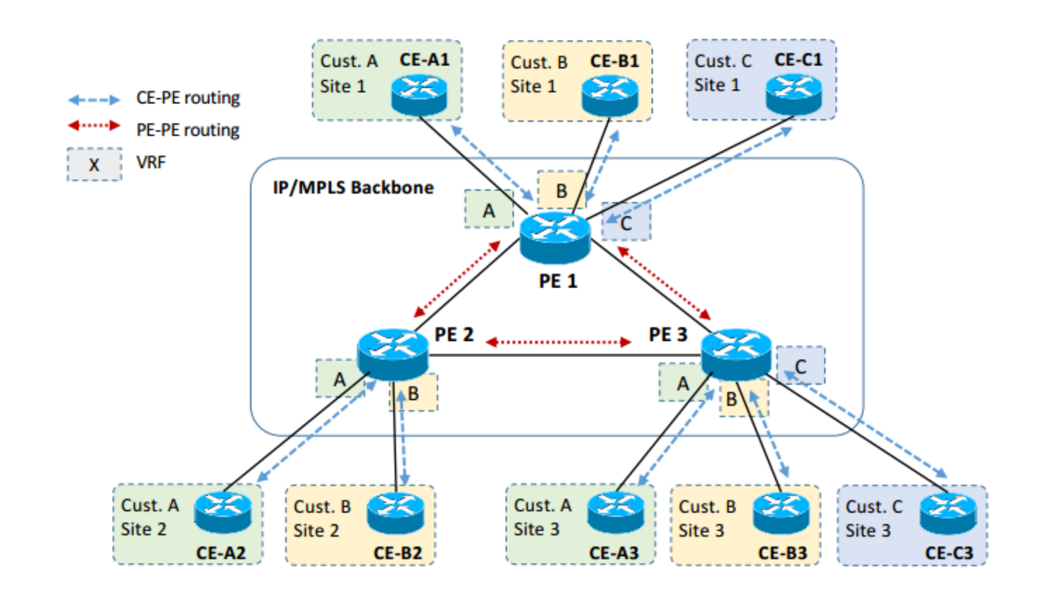


图 1-1 BGP/MPLS L3 VPN 网络拓扑的示例

1.2 研究内容及文章结构

在本文中，我们研究了一种基于图神经网络（GNN）的方法来检测和定位 IP 虚拟专用网络（VPN）中的配置错误。研究了两种 GNN 模型，一种针对客户和提供商边缘路由器之间的路由配置错误，另一种针对不同提供商边缘路由器之间的 VPN 路由错误。

本文共分为五个章节，各个章节内容如下：

第一章绪论。主要介绍本文的研究背景及研究意义、研究内容及各章节安排。

第二章是对网络管理中用到深度学习和图神经网络的概述以及研究现状。

第三章介绍了所研究的两种使用 GNN 来解决 IP VPN 配置的模型。

第四章介绍了两种 GNN 模型的训练及训练集。

第五章实验结果及评价指标。

第 2 章 网络管理及图神经网络

2.1 网络管理的研究现状

近年来，依靠机器学习进行网络管理引起了人们的广泛关注[3]、[4]，文献[5]用不同深度的 CNN 建立了一种实用的监控网络智能管理系统架构，实现了利用 IP 监控网络的真实数据对模型进行评估。

一种能够在基站（BS）进行辐射功率管理的密集无线接入网络（dense-RAN）。在用户流量和可实现速率的约束下，协同管理多基站的辐射功率水平，可以提高网络的长期效能。作者为了解决复杂性和性能之间的权衡问题，使用深度 Q 网络 （DQN） 对具有乘法复杂性约束的多基站能效的整体优化进行建模，以实现接近最优的性能[6]

网络流量管理是现代 VCPS 场景中最大的问题，因为网络资源管理不善可能会降低最终用户的服务质量 (QoS)。为了解决这个问题[7]提出了一种支持软件定义网络（SDN）的方法，名为 SeDaTiVe，它使用深度学习架构来控制 VCPS 环境中网络中的传入流量。在网络流量控制中使用深度学习的优势在于，它可以学习数据包中的隐藏模式，并根据学习到的特征创建最佳路由。较小小区的部署不可避免地会导致更频繁的切换，从而使移动性管理更具挑战性，并减少密集网络部署所提供的容量增益。为了在这种网络环境中充分获得移动用户的收益[8]，提出了一种通过基于深度学习的移动性预测进行移动性管理的智能双连接机制。大多数现有方法应用离散化来逼近实际驾驶条件下的连续最优值，这导致性能相对较低，并且存在离散化误差和维数灾难。

基于 BP 神经网络和 MTLBO 算法，[7]建立了 MTLBO-BP 神经网络预测模型并测试了其性能。通过强化学习公式对投资组合管理进行建模的概念是新颖的，深度 Q 网络最近已成功应用于投资组合管理。引入了一个基于分层深度 QNetwork 的框架，该框架通过减少分配给每个深度 Q 网络的资产数量并将总投资组合价值划分为较小的部分来解决零佣金问题。网络切片和深度强化学习 (DRL) 是实现 5G 和 6G 网络的重要推动因素。使用强化学习 (RL) 和 DRL 算法自主实现每个阶段的方法。

然而，尽管深度学习已近在网络管理中的应用很多，有很多的优秀案例，深度学习很少用于检测网络中的故障，特别是在网络配置中的故障检测 IP VPN 的个数。

2.2 图神经网络的概述

尽管传统的深度学习方法被应用在提取欧氏空间数据的特征方面取得了巨大的成功，但许多实际应用场景中的数据是从非欧式空间生成的，传统的深度学习方法在处理非欧式空间数据上的表现却仍难以使人满意。例如，在电子商务中，一个基于图（Graph）的学习系统能够利用用户和产品之间的交互来做出非常准确的推荐，但图的复杂性使得现有的深度学习算法在处理时面临着巨大的挑战。这是因为图是不规则的，每个图都有一个大小可变的无序节点，图中的每个节点都有不同数量的相邻节点，导致一些重要的操作（例如卷积）在图像（Image）上很容易计算，但不再适合直接用于图。此外，现有深度学习算法的一个核心假设是数据样本之间彼此独立。然而，对于图来说，情况并非如此，图中的每个数据样本（节点）都会有边与图中其他实数据样本（节点）相关，这些信息可用于捕获实例之间的相互依赖关系。

近年来，人们对深度学习方法在图上的扩展越来越感兴趣。在多方因素的成功推动下，研究人员借鉴了卷积网络、循环网络和深度自动编码器的思想，定义和设计了用于处理图数据的神经网络结构，由此一个新的研究热点——“图神经网络（Graph Neural Networks，GNN）”应运而生。

为了能够明确表示 VPN 中涉及的不同路由器之间的关系，有人开始研究图神经网络 (GNN)，这是一种专用于图结构化数据的神经网络，最近受到了广泛关注 [9]、[10]。 GNN 模型由分布在一组层上的多个计算模块组成。图卷积网络（GCN）是用于在节点之间传播信息的最著名的计算模块之一[11]。图的每个节点都嵌入了一组特征，这些特征在每次迭代时与其邻居的特征聚合。还存在其他卷积层，例如边缘条件卷积（ECC）层，其中可以将边缘标签添加到图中以调节邻居之间的信息扩散[12]。设计 GNN 模型需要一系列步骤，首先指定输入图的结构和类型，然后定义学习任务。根据问题的表述，可以有多种类型的学习任务。学习任务的示例是节点分类（例如，学习节点的新特征）或图分类（例如，推断图本身的属性），如图 2-2 所示。在我们的上下文中，将每个 VPN 建模为 图形将使我们能够捕获 VPN 的逻辑拓扑，这对配置参数及其依赖性有直接影响。

此外，使用图数据结构比经典数据结构（例如向量、树）提供了更好的表达能力。因此，通过依赖 GNN，我们希望能够针对比之前的工作更复杂的网络以及更多的配置错误 [8]。

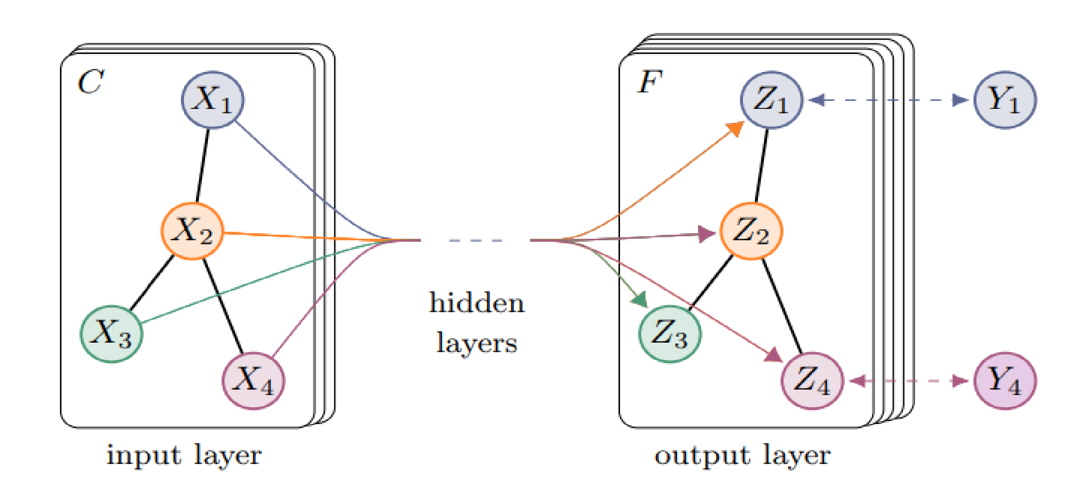


图 2-2 GNN 模型图

第 3 章 L3 VPN 配置数据模型

3.1 两种 GNN 解决 VPN 配置的模型

当使用图神经网络（GNN）来解决 VPN 配置问题时，两个主要挑战是处理不同站点之间的配置参数和策略，以及客户子网的集成。这涉及到节点特征的处理和配置参数之间的相互依赖性。

首先，对于节点特征的处理，GNN 要求所有节点的特征数量和类型相同。考虑到设备可能具有不同的配置参数，可以采取一些方法来解决这个问题。一种方法是通过特征对齐，将不同配置参数的设备映射到相同的特征空间。这可以通过特征转换网络或嵌入层来实现。另一种方法是采用分层模型，为每个设备类型设计一个独立的 GNN 子模型，使每个子模型专注于特定设备类型的配置参数。

其次，配置参数之间的相互依赖性是另一个重要考虑因素。在图结构设计方面，可以使用合适的图结构来表示配置参数之间的依赖关系，例如使用不同类型的边表示不同类型的依赖关系。为了更好地捕获这些关系，可以引入子图级别的注意力机制，使模型能够专注于处理特定类型设备的配置参数。

在解决故障识别任务的配置时，可以考虑将整体任务分解为多个子任务，每个子任务关注于特定类型的设备或配置参数。这有助于降低学习任务的复杂性。同时，使用多个 GNN 模型，并通过融合这些模型的输出来得到对整体配置问题的综合理解，也是一个有效的策略。

综合来说，通过特征对齐、分层模型、图结构设计和任务分割等方法，可以在 GNN 中更有效地处理 VPN 配置问题中的多样性配置参数和复杂的相互依赖性。

事实上，在 GNN 中，所有节点的特征数量和类型必须相同，而对于具有不同配置参数的 PE 和 CE 设备则并非如此。

选择两种不同模型的另一个主要原因与识别故障时配置参数之间的相互依赖性有关。将所有类型故障的所有配置参数合并到同一模型中将再次增加学习任务的复杂性。例如，影响 CE 和 PE 路由器之间数据转发的故障仅取决于这两个设备的配置。相反，影响特定客户站点间数据转发的故障取决于该客户 VPN 涉及的 PE 路由器上部署的所有 VRF。

3.2PE-PE 路由模型

这个模型的目标是使用图神经网络（GNN）来检测和识别 VPN 配置中的错误。在这个模型中，每个图表示一个 VPN，图的节点代表不同的 PE 路由器，而边表示这些路由器之间的连接。

我们关注的错误包括在 PE 路由器上未创建 VRF，或者在 VRF 上配置参数错误，如路由标识符（RD）和路由目标（RT）。此外，对于非全网状 VPN 设置，我们还关注路由策略参数的错误。

每个节点都嵌入了有关该节点上 VPN 配置的信息，而图的边缘上使用二进制特征来指示两个节点之间是否应该进行通信。这种图表示使得 GNN 能够学习节点之间的关系和拓扑信息，从而检测和识别配置错误。

总体而言，这个方法通过将网络配置表示为图，并利用图神经网络的强大能力来处理和理解复杂的网络拓扑和配置关系，从而提高了检测配置错误的效率。

通过这个模型，我们的目标是进行节点分类。对于每个图（即每个 VPN），我们将为每个节点（即每个 PE 路由器）提供一个输出，指示该节点上是否存在配置错误。由于我们想要查明错误，因此每个节点的输出是多个类的向量，每个类代表一个可能的错误。这种学习方法称为监督多类节点分类。在我们的上下文中，这种类型的分类使得可以为每个 VPN 检测和定位包含 VPN 配置错误的 PE 路由器，并使用节点的输出向量精确识别它们。

Configuration error	**Node**
VRF missing	PE
Route Distinguisher (RD)	PE
Import Route Target (RT)	PE
Export Route Target (RT)	PE
Match subnet on import routing policy	PE
Match RT on import routing policy	PE
Match subnet on export routing policy	PE
Apply RT on export routing policy	PE

表 3-1PE-PE 模型配置故障

表 3-1 列出了我们考虑的不同错误。请注意，无论 VPN 类型如何，都必须始终配置 VRF、RD 和 RT 参数，而路由策略参数仅针对非全网状 VPN 设置。

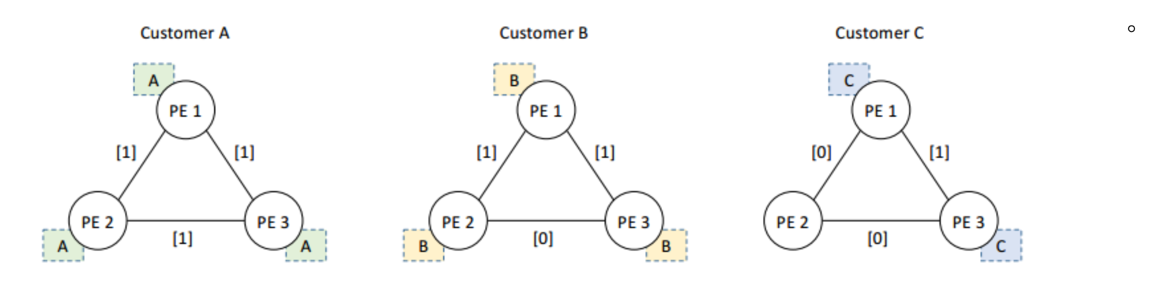


图 3-1PE-PE 的三个 VPN 的三个图结构图

3.3CE-PE 路由模型

这个模型的任务是针对每个 CE-PE 连接检测和识别可能影响 CE 和 PE 路由器之间数据转发的配置错误。CE 指的是 Customer Edge，PE 指的是 Provider Edge。这两个组件之间的路由可以是静态的，也可以依赖路由协议。表 3-2 列举了考虑的到所有可能的故障情况。这些错误大概为四类：VRF（虚拟路由转发）、接口、eBGP 和静态路由。

这个模型的方法是将每个 CE-PE 连接表示为单独的图，其中每个图包含两个节点，一个代表 CE 路由器，另一个代表 PE 路由器。每个节点的功能指定了在添加、更新或删除 VPN 站点时可以设置或修改的配置参数。图 3-2 显示了 BGP/MPLS L3 VPN 网络拓扑的示例中 CE-PE 连接的图表结构。

与之前的模型不同，这里的方法是为每个图而不是每个节点定义输出，实现图分类。相比每个节点有一个输出的方式，这种方法减少了不必要的复杂性。因此，该学习方法采用了有监督的多类图分类。每个类别代表了 CE-PE 连接上可能存在的故障，比如接口上的 IP 地址配置错误或 eBGP 会话上的问题。

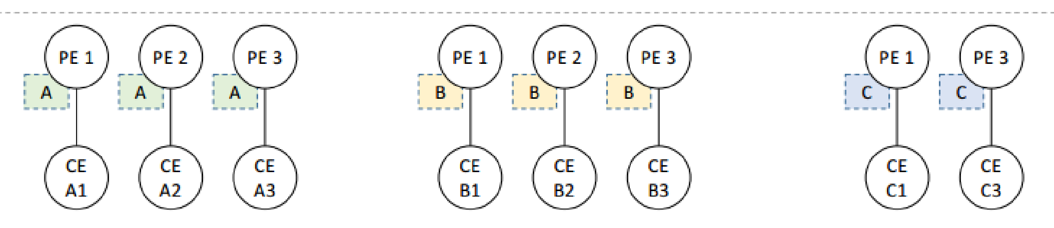


图 3-2 CE-PE 的三个 VPN 的三个图结构图

这种方法的目标是通过学习 CE-PE 连接的图表示来检测和识别配置错误。通过输出向量中的类别标签，模型可以指示特定连接上存在的问题类型，帮助网络管理人员及时纠正配置错误，提高整个网络的稳定性和可维护性。

Category	**Configuration error**	**Node**
VRF	Missing	PE
	RD	PE
Interfaces	IP address	CE/PE
	Mask	CE/PE
	VRF assignation	PE
eBGP	Missing	CE/PE
	Local AS number	CE/PE
	Neighbor IP address	CE/PE
	Neighbor AS number	CE/PE
	Subnet Announcement	CE
	IP forwarding	CE/PE
Static routing	Routes redistribution	PE
	VRF assignation	PE
	Route destination	CE/PE
	Route next hop	CE/PE

表 3-2 CE-PE 模型配置故障

第 4 章 两种模型的训练

4.1 训练集

为了训练和评估 PE-PE 和 CE-PE 模型，使用了[2]根据 IMS Networks 客户的 VPN 配置构建了两个数据集。IMS Networks 在其网络上大约有数百个 VPN，每个 VPN 包含分布在 20 个提供商边缘路由器上的 10 到 30 个远程站点。这些 VPN 大部分都是全网状 VPN。但是，某些 VPN 具有中心辐射型拓扑，需要特定的路由策略（以允许 VPN 的所有站点仅与称为中心的唯一中心站点进行通信）。由于监督机器学习方法需要大量标记数据，必须通过在网络主干上生成随机 VPN 来生成更多配置示例。为了获得平衡的数据集，生成了 50%全网状 VPN 和 50%中心辐射型 VPN。对于配置路由策略的 PE-PE 模型，需要执行此操作。在训练集中，不能简简单单地往里面注入简单的错误，而是在其中加入了随机错误来模拟现实中的场景。

最后，获得了两种类型的 1000 个 IP VPN 配置：全网状和中心辐射型；包含正确和不正确的配置。每个 VPN 有 10 到 30 个 CE 路由器，分布在 20 个 PE 路由器上。这些配置用于创建两个数据集来训练和评估 PE-PE 和 CE-PE 路由 GNN 模型。因此，PE-PE 路由数据集有 20 个节点的 1000 个图（每个 VPN 一个），每个节点根据它们所属的类别进行标记。CE-PE 路由数据集包含 20000 多个标记图，对应于所有 VPN 的所有 CE-PE 连接的总和。对于这两个模型，使用 70%的数据进行训练，10%用于验证，20%用于测试。以上数据集中的节点的类别都和两个表中有所对应。

4.2GNN 两种模型

Spektral 框架被用于构建两个 GNN 的模型：PE-PE 路由模型和 CE-PE 路由模型。这两个模型都采用 Keras API 和 TensorFlow 2，目的是为了提供一个灵活的框架。

PE-PE 路由模型的学习任务是节点分类。模型结构包括两个边缘条件卷积（ECC）层。第一个 ECC 层聚合每个节点的特征及其邻居节点的特征，而第二个 ECC 层作为输出层用于对每个节点进行分类。选择 ECC 层是因为 PE-PE 图中存在边缘特征，它们决定了节点之间的信息通信，而 ECC 层有助于处理这些边缘特征，从而实现准确的节点分类。

CE-PE 路由模型包含三个层次。首先是用于聚合节点特征的 Graph Convolutional Network (GCN)层。接下来是一个池化层，采用全局总和池层，从节点信息中提取每个图的高级表示。最后，使用 Keras 的密集层来计算每个图的输出，完成图分类任务。

总体而言，这两个模型都在 Spektral 框架中使用了不同的卷积层和池化层，以适应各自的学习任务。PE-PE 路由模型专注于节点分类，而 CE-PE 路由模型则处理图分类任务。

第 5 章 GNN 模型性能评估

5.1 参数及评价指标

图表个数为 300，每个图表代表随机分布在 20 个 PE 路由器上的 10 到 30 个远程站点（即 CE 路由器）的 VPN。

$$Precision = \frac{TP}{TP + FP} \quad Recall = \frac{TP}{TP + FN}$$

其中，$Precision$表示计算测精度，$Recall$表示表示召回率，TP 表示预测正确的个数，FP 是错将其他类预测为本类，FN 是本类标签预测为其他类型。

使用 F1 分数来作为评估指标，是精确率和召回率的调和平均数，最大为 1，最小为 0。

$$FI - score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

使用 80% 的 PE-PE 和 CE-PE 数据集进行训练和验证。利用剩下的 20%，我们获得 PE-PE 路由模型的一般 F1 分数值（即所有类别的 F1 分数的加权平均值）为 94%，CE-PE 路由模型为 98%。

5.1 仿真评价

5.1.1PE-PE 模型

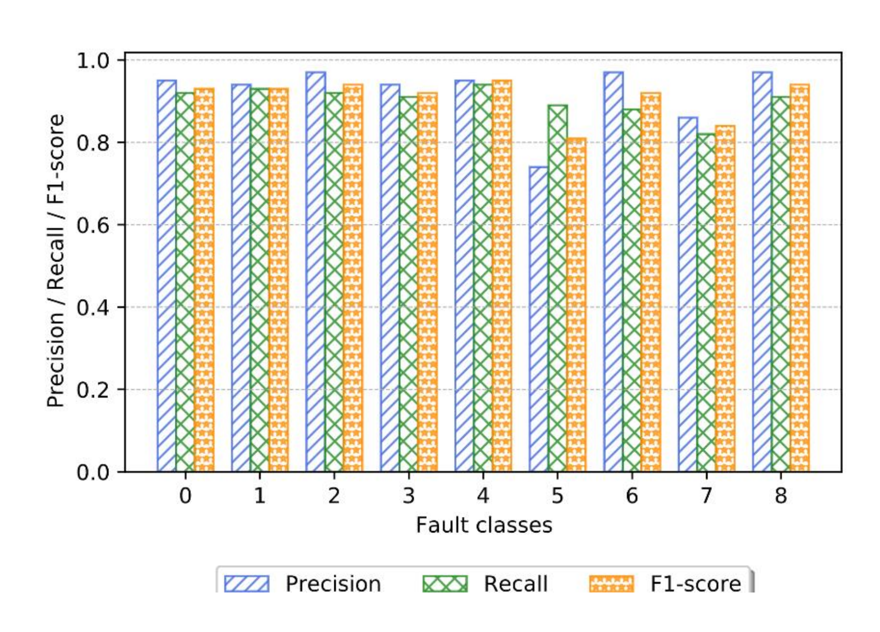


图 5-1PE-PE 每个故障类型的性能图

图 5-1 绘制了使用包含 300 个图表的数据集计算的精确度、召回率和 F1 分数值，每个图表代表随机分布在 20 个 PE 路由器上的 10 到 30 个 CE 路由器的 VPN。它总共提供了 6000 个 PE 节点进行分类。1 至 8 类对应于表 3-1 中列出的配置错误，0 类表示节点上没有错误。除了第 5 和第 7 类的 F1 分数接近 80%，其他的 F1 分数值均接近 90%。

5 类和 7 类表示应分别在导入和导出路由策略上配置的子网上的错误。这两个参数用于中心辐射型 VPN，以过滤要导入到客户路由表中或从客户路由表中导出的路由。该模型对于这些类别的表现不佳主要有两个原因。首先，该子网参数在数据集中没有可以比较或关联的其他特征（它仅出现在 CE-PE 模型中客户边缘路由器的其他位置）。其次，训练数据集对于这些类型的错误是不平衡的。事实上，中心辐射型 VPN 占训练数据集中 VPN 的 50%，这使得路由策略错误的代表性低于任何 VPN 上可能发生的其他错误。

6 类和 8 类也是路由策略上的错误，但重点关注路由目标(RT)值（在导入策略上匹配并在导出策略上设置）。与具有子网的 5 类和 7 类相反，该 RT 值与同一 PE 节点上客户 VRF 中存在的 RT 值相关。

结果表明，该模型可以学习表示 VRF 的特征与路由策略参数之间存在的关系。

图 5-2 所示的条形图证实了之前的结果。一般精度、召回率和 F1 分数值是通过在三种不同的数据集上测试 PE-PE 模型来计算的：一种仅由全网状 VPN 组成，一种仅由中心辐射型 VPN 组成，一种由两种类型组成。

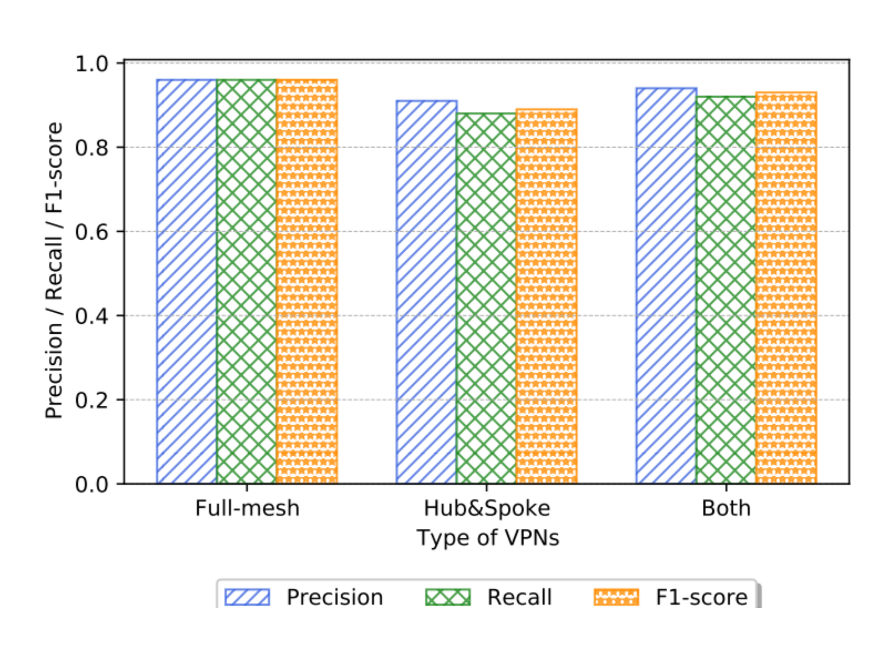


图 5-2 每种 VPN 类型的详细性能

该图表可以清楚地看出，该模型在全网状 VPN（即没有路由策略）下学习效果更好。

5.1.1CE-PE 模型

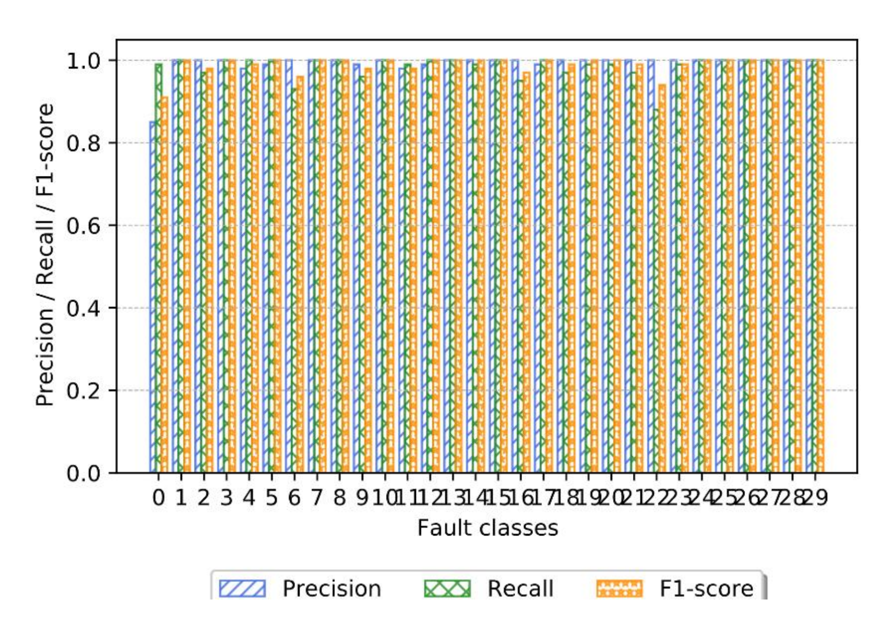


图 5-3 CE-PE 每个故障类型的性能图

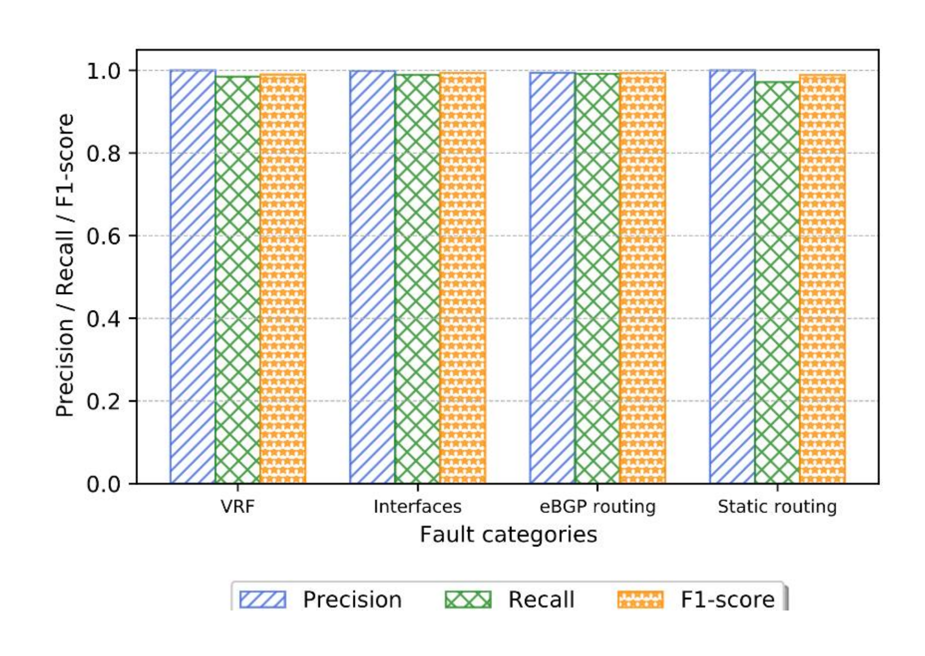


图 5-4 CE-PE 故障分类后的性能图

这些测试是在用于绘制图 5-1 中的图表的相同数据集上进行的。由于该模型中有多种类型的错误（30 类），为了更好的可读性，我们选择根据它们所属的类别对它们进行分组。

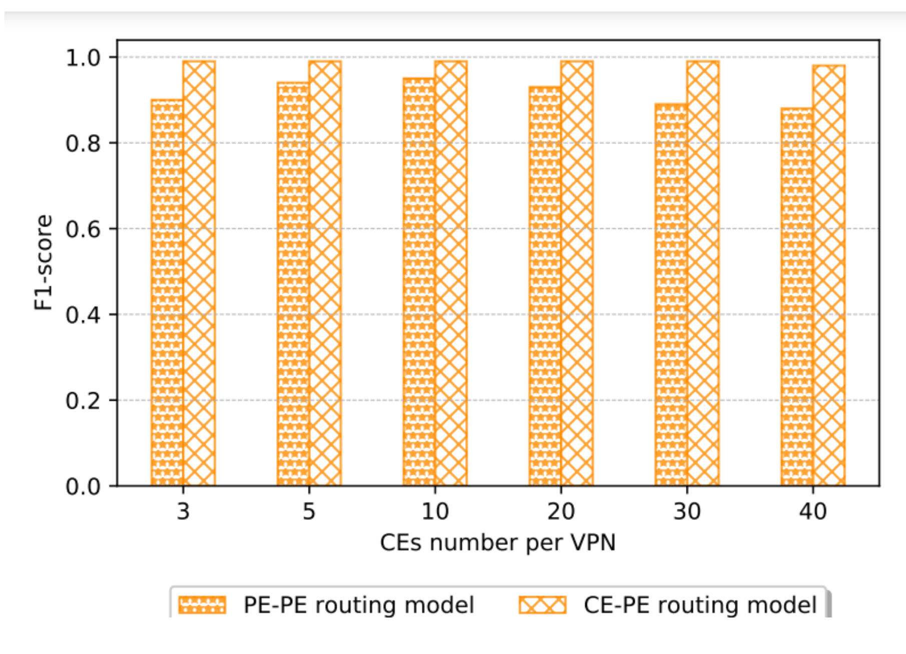


图 5-5 VPN 中不同站点的 F1 分数

图 5 显示了使用 7 个每个都包含 300 个 VPN 数据集的 PE-PE 和 CE-PE 路由模型的 F1 分数。我们可以观察到，改变每个 VPN 的 CE 数量不会影响 CE-PE 路由模型。这是有道理的，因为改变 CE 的数量不会改变 CE-PE 图上的任何内容，它只会改变要分类的图的数量。对于 PE-PE 路由模型，每个 VPN10、20 和 30 个 CE 的全局 F1 分数与图 3 中绘制的平均 F1 分数相似（大约 90%）。这个结果是一致的，因为该模型已经使用包含具有 10 到 30 个 CE 路由器的 VPN 的数据集进行了训练。更有趣的结果是具有 3、5、40 和 50 个客户路由器的 VPN 的 F1 分数。尽管训练数据集中不存在这些 VPN 大小，但分类精度非常好（F1 分数仅降低 3%或 4%），表明学习过程泛化良好，没有过度拟合。

结论

在本文中，研究了一种基于图神经网络的方法来检测和定位 IP VPN 中的配置错误。

文章的研究包含两个不同的 GNN 模型，每个模型处理不同的分类问题。PE-PE 模型执行节点分类，以针对每个图预测 VRF 表或路由策略上可能的配置错误。CE-PE 模型执行图分类以预测客户-提供商路由上可能的配置错误。

按错误类型、VPN 拓扑和 VPN 大小排序的不同数据集的测试结果表明，检测配置错误的预测准确度在 80% 到 90% 之间。此外，对训练数据集中不存在的 VPN 模式（即具有不同数量的站点以及提供商边缘路由器上的不同位置）进行的测试表明性能仅下降 4%，这意味着模型在未见过的数据上表现良好。

参考文献

[1] E. Rosen and Y. Rekhter, “BGP/MPLS IP Virtual Private Networks(VPNs),” RFC Editor, RFC 4364, Feb. 2006

[2]E. -H. Mohammedi, E. Lavinal and G. Fleury, "Detecting and locating configuration errors in IP VPNs with Graph Neural Networks," NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium, Budapest, Hungary, 2022, pp. 1-6.

[3] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, “Machine learning for networking: Workflow, advances and opportunities,” IEEE Network, vol. 32, no. 2, p. 92–99, Mar. 2018.

[4] R. Boutaba, M. A. Salahuddin, N. Limam, S. Ayoubi, N. Shahriar, and et al., “A comprehensive survey on machine learning for networking:evolution, applications and research opportunities,” Journal of Internet Services and Applications, vol. 9, no. 16, Jun. 2018.

[5]C. -C. Kao, Y. -C. Lai, J. Pei, C. -W. Chang, F. -H. Kuo and J. -Y. Shun, "Intelligent Management System: Deep Convolutional Neural Networks for Automatic Attribute Recognition in IP Surveillance Networks," 2020 21st Asia-Pacific Network Operations and Management Symposium (APNOMS), Daegu, Korea (South), 2020, pp. 397-400.

[6]Y. Chang et al., "Collaborative Multi-BS Power Management for Dense Radio Access Network Using Deep Reinforcement Learning," in IEEE Transactions on Green Communications and Networking, vol. 7, no. 4, pp. 2104-2116, Dec. 2023.

[7]A. Jindal, G. S. Aujla, N. Kumar, R. Chaudhary, M. S. Obaidat and I. You, "SeDaTiVe: SDN-Enabled Deep Learning Architecture for Network Traffic Control in Vehicular Cyber-Physical Systems," in IEEE Network, vol. 32, no. 6, pp. 66-73, November/December 2018.

[8]C. Wang, Z. Zhao, Q. Sun and H. Zhang, "Deep Learning-Based Intelligent Dual Connectivity for Mobility Management in Dense Network," 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall), Chicago, IL, USA, 2018, pp. 1-5.

[9] Z. Liu and J. Zhou, “Introduction to graph neural networks,” Synthesis Lectures on Artificial Intelligence and Machine Learning, vol. 14, no. 2,pp. 1–127, 2020.

[10] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, “Graph neural networks: A review of methods and applications,”AI Open, vol. 1, pp. 57–81, 2020.

[11] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” CoRR, vol. abs/1609.02907, 2016.

[12] M. Simonovsky and N. Komodakis, “Dynamic Edge-Conditioned Filters in Convolutional Neural Networks on Graphs,” in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.